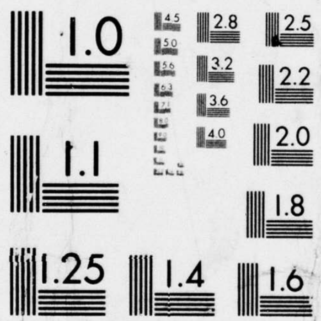
AD-A031 051 COLORADO STATE UNIV FORT COLLINS DEPT OF ELECTRICAL --ETC F/G 12/1
THE SOLUTION OF A SPECIAL SET OF HERMITIAN TOEPLITZ LINEAR EQUA--ETC(U)
JAN 75 D C FARDEN N00014-67-A-0299-0019

UNCLASSIFIED TR-12(ONR) NL

1 OF 1
AD
A031051

END
DATE
FILMED
11-76



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

Good DDC FG.

1

The Solution of a Special
Set of Hermitian Toeplitz Linear Equations

by

David C. Farden

ONR Technical Report #12

August 1975

Prepared for the Office of Naval Research
under Contract N00014-75-C-0518

L. L. Scharf and M. M. Siddiqui, Principal Investigators

6 The Solution of a Special
Set of Hermitian Toeplitz Linear Equations.

by

9 Technical rept.,

10 David C. Farden

Electrical Engineering Department
Colorado State University
Fort Collins, Colorado 80523

14 TR-12(ONR)

11 January 1975

12 13p.

Submitted to

ACM Transactions on
Mathematical Software

15

This work was supported by the Office of Naval Research under Grants N00014-67-A-0299-0019, and N00014-75-C-0518, and the Naval Undersea Center under Contracts N66001-74-C-0035 and N66001-75-C-0224.

406 434

ABSTRACT

The solution of a set of m linear equations $L_m s_m = d_m$, where L_m is an mth order Hermitian Toeplitz matrix and the elements of d_m possess a Hermitian symmetry, is considered. A specialized algorithm is developed for this case which solves for s_m in approximately $1.5m^2$ "operations," whereas the Hermitian case of an algorithm developed by Zohar solves for s_m in approximately $2m^2$ "operations." An "operation" is used here to denote one addition and one multiplication. A further reduction in computational requirements is shown in case L_m and d_m are real. As with Zohar's algorithm, the specialized algorithm requires that all principal minors of L_m be nonzero.

KEY WORDS AND PHRASES: Linear algebra, linear equations, Toeplitz matrix, computer programming

CR CATEGORIES: 5.14, 5.25

1. Introduction

Consider the set of linear equations

$$L_m s_m = d_m. \tag{1}$$

Zohar [1] makes use of the Trench algorithm [2], [3] to develop an efficient algorithm for solving (1) when s_m, d_m are mx1 matrices and L_m is a non-Hermitian mth-order Toeplitz matrix. In this paper, an efficient algorithm is developed for solving (1) when L_m is a Hermitian Toeplitz matrix and d_m satisfies

$$d_m^* = \hat{d}_m , \tag{2}$$

where the symbol ^ is used to denote the reversed ordering of the elements of d_m, i.e., $(\hat{d}_m)_{i,1} = (d_m)_{m+1-i,1}$ and * denotes complex conjugate. Such a specialized case can arise, for example, in the design of digital filters, as discussed in [4]. The following example serves to illustrate how such a system of equations can arise.

EXAMPLE. Let $\alpha(t)$, $\beta(t)$, $\gamma(t)$ be jointly wide-sense stationary complex-valued stochastic processes with $\alpha(t) = \beta(t) + \gamma(t)$, where $E\{\beta(t)\gamma^*(s)\} = 0$ for all real t and s and $E\{\cdot\}$ denotes statistical expectation. On the basis of the observation vector $a_m(k)$, $\tilde{a}_m(k) = [\alpha(k) \quad \alpha(k-1) \cdots \alpha(k-m+1)]$, where the symbol ~ denotes matrix transpose, it is desired to compute a linear minimum mean-square error (MMSE) estimate of $\beta(k-p)$, i.e., it is desired to minimize the quantity $E\{|\tilde{s}_m a_m(k) - \beta(k-p)|^2\}$ with respect to s_m. It is easily shown that the desired solution, s_m, satisfies (1), with $L_m = E\{a_m^*(k)\tilde{a}_m(k)\}$ and $d_m = E\{\beta(k-p)a_m^*(k)\}$. Since $\alpha(t)$ is wide-sense stationary, L_m is a Hermitian Toeplitz matrix. With $p = (m+1)/2$, it is easily seen that (2) is satisfied since $\beta(t)$ and $\gamma(t)$ are jointly wide-sense stationary..

A useful consequence of the assumptions that $d_m^* = \hat{d}_m$ and $L_m^* = \tilde{L}_m$ is that $s_m^* = \hat{s}_m$. Define E_m to be the mxm exchange matrix of Zohar [3], i.e., $E_m a_m = \hat{a}_m$ for any mx1 matrix a_m. Note that $E_m E_m = I_m$, where I_m is the mxm identity matrix. Since L_m is persymmetric [3], $E_m L_m E_m = \tilde{L}_m$. Since $\hat{d}_m = d_m^*$, from (1) we have $E_m L_m (E_m E_m) s_m = L_m^* s_m^*$, so that $\tilde{L}_m \hat{s}_m = L_m^* s_m^*$, i.e., $\hat{s}_m = s_m^*$.

The specialized algorithm developed in Section 3 of this paper solves (1) with d_m satisfying (2) in approximately $1.5m^2$ complex "operations," whereas the Hermitian case of Zohar's algorithm [1] uses approximately $2m^2$ complex "operations." An "operation" is used here to denote one addition and one multiplication. In case L_m,d_m (and hence s_m) are real, the results of Section 3 can be used to solve (1) in approximately $1.25m^2$ real multiplications and $1.5m^2$ real additions.

Both Zohar's algorithm [1] and the specialized algorithm developed in Section 3 make use of Phase 1 of the Trench algorithm [1]-[3]. Rather than review the results necessary for the development of Section 3, it is assumed that the reader is familiar with the work of Zohar [3].

2. Preliminaries

Since the techniques used in this paper are inherently related to those used by Zohar [1], an attempt is made to follow the same notational conventions. Greek letters are used for scalars, capital letters for square matrices, and lower-case letters for column matrices. Subscripts used on matrices are used to denote the number of elements in one column of the matrix.

Since Phase 1 of the Trench algorithm requires that all principal minors of L_m be nonzero, it is assumed that (1) has been normalized so that L_m has ones along its main diagonal.

3. The Specialized Algorithm

Consider the system of equations $L_m s_m = d_m$, where L_m is an mth order normalized Hermitian Toeplitz matrix and $d_m^* = \hat{d}_m$, so that d_m may be written as $\tilde{d}_m = [\xi_{\frac{m+1}{2}} \cdots \xi_2\ \xi_1\ \xi_2^* \cdots \xi_{\frac{m+1}{2}}^*]$ for m odd and

$\tilde{d}_m = [\ \xi_{\frac{m}{2}} \cdots \xi_2\ \xi_1\ \xi_1^* \cdots \xi_{\frac{m}{2}}^*]$ for m even. For m even or odd we may write

$\tilde{d}_{i+2} = [\xi_{[\frac{i+3}{2}]}\ \tilde{d}_i\ \xi^*_{[\frac{i+3}{2}]}]$, for $i = 1, 2, \cdots, m-2$ where $[x]$ denotes the largest integer less than or equal to x. The Hermitian Toeplitz nature of L_m enables us to write

$$L_{i+2} = \begin{bmatrix} 1 & \tilde{r}_{i+1} \\ r^*_{i+1} & L_{i+1} \end{bmatrix} = \begin{bmatrix} L_{i+1} & \hat{r}_{i+1} \\ \tilde{\hat{r}}^*_{i+1} & 1 \end{bmatrix}, \qquad (3)$$

where $\tilde{r}_{i+1} = [\rho_1 \rho_2 \cdots \rho_{i+1}]$ $(0 \le i \le m-2)$. Clearly, (3) may be rewritten as

$$L_{i+2} = \begin{bmatrix} 1 & \tilde{r}_i & \\ r_i^* & L_i & \hat{r}_{i+1} \\ & \tilde{\hat{r}}^*_{i+1} & 1 \end{bmatrix}.$$

Defining $L_{i+2}\, s_{i+2} = d_{i+2}$ $(1 \le i \le m-2)$, we have $L_{i+2}\left\{ s_{i+2} - \begin{bmatrix} 0 \\ s_i \\ 0 \end{bmatrix} \right\} = \begin{bmatrix} \theta_i \\ 0_i \\ \theta_i^* \end{bmatrix}$,

where $\theta_i = \xi_{[\frac{i+3}{2}]} - \tilde{r}_i s_i$ and 0_i is an ix1 column matrix of zeros.

Defining $B_{i+2}=L_{i+2}^{-1}$, we obtain

$$s_{i+2} = \begin{bmatrix} 0 \\ s_i \\ 0 \end{bmatrix} + B_{i+2} \begin{bmatrix} \theta_i \\ 0_i \\ \theta_i^* \end{bmatrix} \qquad (4)$$

Since the inverse of a Hermitian persymmetric matrix is a Hermitian persymmetric matrix [3], B_{i+2} may be expressed in the form

$$B_{i+2} = \lambda_{i+1}^{-1} \begin{bmatrix} 1 & \tilde{e}_{i+1} \\ e_{i+1}^* & M_{i+1} \end{bmatrix} = \lambda_{i+1}^{-1} \begin{bmatrix} P_{i+1} & \hat{e}_{i+1} \\ \hat{\tilde{e}}_{i+1}^* & 1 \end{bmatrix} .$$

Letting $f_i = [I_i \; 0_i] \, e_{i+1}$, we may write

$$B_{i+2} = \lambda_{i+1}^{-1} \begin{bmatrix} 1 & \tilde{f}_i & \\ f_i^* & Q_i & \hat{e}_{i+1} \\ & \hat{\tilde{e}}_{i+1}^* & 1 \end{bmatrix} . \qquad (5)$$

Substituting (5) into (4) we obtain the result

$$s_{i+2} = \begin{bmatrix} 0 \\ s_i \\ 0 \end{bmatrix} + \lambda_{i+1}^{-1}\theta_i \left\{ \begin{bmatrix} 1 \\ e_{i+1}^* \end{bmatrix} + \theta_i^* \theta_i^{-1} \begin{bmatrix} \hat{e}_{i+1} \\ 1 \end{bmatrix} \right\} . \qquad (6)$$

In order to make use of this result, we apply the recursive relationships for Phase 1 of the Trench algorithm [1]:

Initial values: $e_1 = -\rho_1$, $\lambda_1 = 1 - |\rho_1|^2$

Recursive relationships: $\eta_i = -\rho_{i+1} - \tilde{e}_i \hat{r}_i$,

$$e_{i+1} = \begin{bmatrix} e_i + \eta_i \lambda_i^{-1} \hat{e}_i^{*} \\ \\ \eta_i \lambda_i^{-1} \end{bmatrix}, \quad \lambda_{i+1} = \lambda_i - |\eta_i|^2 \lambda_i^{-1} .$$

Finally, Phase 1 of the Trench algorithm and (6) may be combined by noting that

$$s_1 = \xi_1 \tag{7}$$

and

$$s_2 = (1 - |\rho_1|^2)^{-1} \begin{bmatrix} \xi_1 - \rho_1 \xi_1^{*} \\ \\ \xi_1^{*} - \rho_1^{*} \xi_1 \end{bmatrix} . \tag{8}$$

An immediate consequence of (6), (7), and (8) is that $s_{i+2}^{*} = \hat{s}_{i+2}$ since λ_{i+1} is real-valued. Consequently, there are two sources of increased computational speed in the specialized algorithm: (i) s_{i+2} need only be computed for $i=1,3,5,\cdots,m-2$ when m is odd and for $i=2,4,6,\cdots,m-2$ when m is even, and (ii) approximately half ($[\frac{i+3}{2}]$) of the elements of s_{i+2} need to be computed using (6), the remaining elements being obtained from the relationship $s_{i+2}^{*} = \hat{s}_{i+2}$. The following is a summary of the algorithm.

PROBLEM FORMULATION: $L_m s_m = d_m, L_m = \begin{bmatrix} 1 & \tilde{r}_{m-1} \\ \\ r_{m-1}^{*} & L_{m-1} \end{bmatrix}$,

$\tilde{r}_i = [\rho_1 \rho_2 \cdots \rho_i]$ $(1 \leq i \leq m-1)$,

$\tilde{d}_{i+2} = [\ \xi_{[\frac{i+3}{2}]} \ \tilde{d}_i \ \xi^{*}_{[\frac{i+3}{2}]}\ ]$, $s_m = ?$

Initial values: $e_1 = -\rho_1$, $\lambda_1 = 1-|\rho_1|^2$,

$$s_1 = \xi_1 \ , s_2 = \lambda_1^{-1} \begin{bmatrix} \xi_1 - \rho_1 \ \xi_1^* \\ \\ \xi_1^* - \rho_1^* \ \xi_1 \end{bmatrix}$$

Recursive relations: Compute η_i, e_{i+1}, and λ_{i+1} for $i=1,2,\cdots, m-2$. Compute θ_i and s_{i+2} for $i = 1,3,5,\cdots, m-2$ for m odd and $i = 2,4,6,\cdots, m-2$ for m even.

$$\eta_i = -\rho_{i+1} - \tilde{e}_i \hat{r}_i$$

$$e_{i+1} = \begin{bmatrix} e_i + \eta_i \lambda_i^{-1} \hat{e}_i^* \\ \\ \eta_i \lambda_i^{-1} \end{bmatrix}$$

$$\lambda_{i+1} = \lambda_i - |\eta_i|^2 \lambda_i^{-1}$$

$$\theta_i = \xi_{[\frac{i+3}{2}]} - \tilde{r}_i s_i$$

$$s_{i+2} = \begin{bmatrix} 0 \\ s_i \\ 0 \end{bmatrix} + \lambda_{i+1}^{-1} \theta_i \left\{ \begin{bmatrix} 1 \\ e_{i+1}^* \end{bmatrix} + \theta_i^* \theta_i^{-1} \begin{bmatrix} \hat{e}_{i+1} \\ 1 \end{bmatrix} \right\}$$

Making use of the fact that only $[\frac{i+3}{2}]$ elements of s_{i+2} need be computed, the above algorithm requires approximately $1.5m^2$ additions and $1.5m^2$ multiplications for the solution of s_m. This compares with $2m^2$ for the Hermitian case of Zohar's algorithm [1].

In case L_m, d_m (and hence s_m) are real, an even further reduction in computational requirements results. For this case (6) may be rewritten as

$$s_{i+2} = \begin{bmatrix} 0 \\ s_i \\ 0 \end{bmatrix} + \lambda_{i+1}^{-1} \ \theta_i \left\{ \begin{bmatrix} 1 \\ \\ e_{i+1} \end{bmatrix} + \begin{bmatrix} \hat{e}_{i+1} \\ \\ 1 \end{bmatrix} \right\} , \qquad (9)$$

and the computation of $\tilde{r}_i s_i$ in the expression for θ_i may be computed as

$$\tilde{r}_i s_i = \sum_{\ell=1}^{i/2} (s_i)_\ell (\rho_\ell + \rho_{i+1-\ell}) \tag{10}$$

for i even and

$$\tilde{r}_i s_i = \sum_{\ell=1}^{\frac{i-1}{2}} (s_i)_\ell (\rho_\ell + \rho_{i+1-\ell}) + (s_i)_{\frac{i+1}{2}} \rho_{\frac{i+1}{2}} \tag{11}$$

for i odd. Making use of these expressions, the specialized algorithm requires approximately $1.5m^2$ additions and $1.25m^2$ multiplications. A slightly different form of (9) can be easily obtained as

$$s_{i+2} = \begin{bmatrix} 0 \\ s_i \\ 0 \end{bmatrix} + \frac{\theta_i}{\lambda_i - \eta_i} \begin{bmatrix} 1 \\ e_i + \hat{e}_i \\ 1 \end{bmatrix} . \tag{12}$$

This final expression (12) is slightly more efficient than (9). A FORTRAN routine for the specialized algorithm making use of (10)-(12) is presented in [5].

EXAMPLE. Let $\rho_i = (i+1)^{-1}$ for $i=1,2,\cdots, m-1$ and $\xi_i = i^{-1}$, for $i=1,2,\cdots, [\frac{m+1}{2}]$. A FORTRAN routine, called TPSLV, based on the symmetric case of [1] was written for a timing comparison with the FORTRAN routine, called SYMM, presented in [5]. The time needed (in seconds) for each routine to compute s_m for this example with $m \in \{10,50,100,500\}$ is indicated in the following table.

M	TPSLV	SYMM
10	.005	.005
50	.089	.057
100	.343	.217
500	8.266	5.233

The above results, obtained on a CDC 6400 computer, agree with the computational considerations presented above.

4. Concluding Remarks

An algorithm has been developed for the solution of a specialized set of Toeplitz linear equations that arise in linear filtering applications. The savings in computational requirements of the new algorithm over the results of Zohar [1] are approximately 25% for the Hermitian case and 37.5% for the real case. Finally, it is noted that the techniques used in developing the specialized algorithm can indeed be applied to the general case treated by Zohar [1]; however, such a development results in an algorithm having no computational advantage over the generalized algorithm of [1].

REFERENCES

1. ZOHAR, S. The solution of a Toeplitz set of linear equations. *J. ACM 21*, 2 (April 1974), 272-276.

2. TRENCH, W. F. An algorithm for the inversion of finite Toeplitz Matrices. *J. SIAM 12*, 3 (Sept. 1964), 515-522.

3. ZOHAR, S. Toeplitz matrix inversion: The algorithm of W. F. Trench. *J. ACM 16*, 4 (Oct. 1969), 592-601.

4. FARDEN, D. C. and SCHARF, L. L. Statistical design of nonrecursive digital filters. *IEEE Trans. Acoustics, Speech, and Signal Processing. ASSP-22*, 3 (June 1974), 188-196.

5. FARDEN, D. C. and SCHARF, L. L. Authors' reply to "Comments on 'Statistical design of nonrecursive digital filters'." *IEEE Trans. Acoustics, Speech, and Signal Processing. ASSP-23*, 5 (Oct. 1975).

Unclassified 8-25-75

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 12	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) The Solution of a Special Set of Hermitian Toeplitz Linear Equations		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) David C. Farden		8. CONTRACT OR GRANT NUMBER(s) N00014-75-C-0518
9. PERFORMING ORGANIZATION NAME AND ADDRESS Department of Electrical Engineering Colorado State University Fort Collins, CO 80523		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research, Code 436 Statistics and Probability Branch Arlington, VA 22217		12. REPORT DATE August 1975
		13. NUMBER OF PAGES 10
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Linear algebra; Linear equations; Toeplitz matrix; Computer programming.

sub m

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The solution of a set of m linear equations $L_m s_m = d_m$, where L_m is an mth order Hermitian Toeplitz matrix and the elements of d_m possess a Hermitian symmetry, is considered. A specialized algorithm is developed for this case which solves for s_m in ~~approximately~~ about $1.5m^2$ operations, whereas the Hermitian case of an algorithm developed by Zohar solves for s_m in approximately $2m^2$ → next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified m square 8-25-75

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

sub-m

20.

cont. "operations." An "operation" is used here to denote one addition and one multiplication. A further reduction in computational requirements is shown in case L_m and d_m are real. As with Zohar's algorithm, the specialized algorithm requires that all principal minors of L_m be nonzero.

sub m